

# DECUS

## HOLLAND BULLETIN



DIGITAL EQUIPMENT  
COMPUTER USERS SOCIETY

### DECUS Holland Symposium 1985 succesvol

Nr. 26, augustus 1985

Wij hebben het er dan maar eens op gewaagd. Een twee-daags symposium. Al lang liepen we met het idee. In 1983 voelde 42% van de symposiumbezoekers hiervoor en was 15% tegen; in 1984 kregen we heel andere getallen: 17% voor 2 dagen, 80% voor 1 dag. Maar het bestuur, eigenwijs als ze is, wilde toch doorzetten. Om die 80% nu niet af te schrikken, hadden we wel besloten dat de eerste dag een traditionele dag zou zijn, waar iedereen apart voor kon inschrijven, terwijl de tweede dag een eigen karakter moest gaan dragen. Een karakter dat we in 1984 al hadden uitgetoetst met een parallelle sessie in de middag. Managerieel moest het worden. O.K., nek uitgestoken, zaal gereserveerd, sprekers gezocht, en dan maar hopen dat het zou aanslaan. En het sloeg aan! Wij hebben onze targets gehaald, de inhoud van beide dagen werd hoog gewaardeerd. Wat wil je als bestuur en symposiumorganisatie nog meer? 65% van de deelnemers is alleen de eerste dag gekomen (de oude, traditionele dag blijft dus trekken), 12% kwam alleen de tweede dag en 23% was dus beide dagen aanwezig. Die groep van 12% is eigenlijk best interessant. In het afgelopen jaar hebben we een management SIG opgericht, een nieuw fenomeen binnen DECUS en destijds niet alleen in Nederland, maar zelfs wereldwijd. Die SIG trekt dus blijkbaar een flinke groep mensen die eerder, denk ik, niet zo veel te zoeken hadden binnen DECUS.

Voor ruim 300 personen had onze symposiumcoördinator Sander Kortebout blijkbaar een interessant programma kunnen voorschotelen. De waarderingscijfers van de enquête waren hoog. Zo in het algemeen scores tussen 7 en 8 op een schaal van 10 (wat een precisie vragen wij toch!). En dat de mensen toch echt voor de lezingen komen blijkt uit het feit dat de borrels die wij aan het eind van beide dagen organiseerden, wat magertjes bezocht werden. Toch ben ik van mening dat een DECUS Symposium zo iets nodig heeft. DECUS is nu eenmaal een vereniging van gebruikers waar kennisuitwisseling centraal staat. Juist activiteiten zoals een borrel geven alle leden ruime kans om met elkaar te praten en niet alleen te luisteren naar een paar sprekers.

Verder moet ik hier niet vergeten de zeer goede expositie te vermelden. Digital heeft echt zijn uiterste best gedaan om ons veel apparatuur en software te tonen. Ook waren er specialisten aanwezig om eventuele vragen te beantwoorden.

### Inhoud

- DECUS Holland Symposium 1985 succesvol
- Resultaten enquête DECUS Holland Symposium 1985
- Verslagen van de VAX SIG en RT-11 SIG bijeenkomst op 23 april 1984 in de RAI in Amsterdam
- Hand-outs DECUS Holland Symposium 1985
- Verslag RT-11 SIG bijeenkomst 27 februari 1985
- Oproep aan IAS gebruikers
- Nieuwe programma's in de programmabibliotheek
- Over TRAMP gesproken...
- Prijslijst DECUS Programma-bibliotheek
- Nieuwe VAX SIG tapes. Waar blijven uw bijdragen?
- AI Tools and Techniques: RI, XSEL and PTRANS



Tenslotte nog iets over de accommodatie, want hierover krijgen wij elk jaar wel een paar kritische opmerkingen. Dat de RAI niet een ideale plek is, zijn wij geheel met velen eens. De RAI is wat lastig bereikbaar, de zalen zijn niet allen van even goede kwaliteit, de lunch is matig. Allemaal waar, maar ons probleem is dat we voor de dag waarop de SIG sessies worden gehouden veel zaaltjes nodig hebben, terwijl er voor de plenaire sessies een zaal van zeker 350 personen beschikbaar moet zijn. Als we dan ook nog een expositie-ruimte hebben en alle zalen bij voorkeur dicht bij elkaar moeten liggen, dan is er niet zo gek veel keus in Nederland. Slechts indien we zouden besluiten op ons nationale symposium geen parallelle SIG sessies meer te houden (wie wil dat en wie niet ???), liggen er een flink aantal andere mogelijkheden open,

waardoor we dan ook ieder jaar in een andere plaats bij elkaar zouden kunnen komen.

Ronald Beetz, Voorzitter

P.S. Hierbij wil ik iedereen bedanken die bijgedragen heeft aan het slagen van deze dag en met name Sander Kortenbout, Mieke Lips, Femma Kroes en de bemanning in de expositieruimte.



## Resultaten enquête DECUS Holland Symposium 1985

### 23 april 1985 – Technische Dag

Aantal deelnemers: ca. 300

Aantal geretourneerde enquêteformulieren: 138

Algemene indruk: 7.4	Akkommodatie en lunch: 7.1	Indeling van de dag: 7.6
Bezocht u beide symposiumdagen?	ja: 47      nee: 91	niet ingevuld: —
Welke lokatie heeft uw voorkeur?	RAI Amsterdam : 48 Jaarbeurs Utrecht : 71 Andere lokatie : 5 (Babylon Den Haag, POC E'hoven, centrum v.h. land) Geen voorkeur : 5 Niet ingevuld : 9	
Gemiddelde waardering ochtendlezingen	inhoud : 7.1 presentatie : 7.1	
Gemiddelde waardering middaglezingen	inhoud : 7.3 presentatie : 7.5	

### 24 april 1985 – Management Dag

Aantal deelnemers: ca. 105

Aantal geretourneerde enquêteformulieren: 60

Algemene indruk: 7.6	Akkommodatie en lunch: 7.5	Indeling van de dag: 7.9
Bezocht u beide symposiumdagen?	ja: 37      nee: 21	niet ingevuld: 2
Welke lokatie heeft uw voorkeur?	RAI Amsterdam : 27 Jaarbeurs Utrecht : 20 Andere lokatie : 2 (Babylon Den Haag / kleinere accommodatie) Geen voorkeur : 4 Niet ingevuld : 7	
Gemiddelde waardering van de lezingen	inhoud : 7.4 presentatie : 7.7	





*Gezellige drukte bij de registratie-balie*





# Hand-outs DECUS Holland Symposium 1985

Voor belangstellenden is op aanvraag een kopie van de overhead slides beschikbaar van de volgende lezingen die gehouden werden tijdens het DECUS Holland Symposium:

- Het ontwerp van een Local Area Network in een chemische industrie  
F.P. Schoot, Dow Chemical Terneuzen
- KERMIT  
C.J. de Groot, Landbouwhogeschool Wageningen
- Breedband netwerken  
A. Langen, TH Eindhoven
- EARN, European Academic Research Network en zijn toepassingen voor VAX/VMS gebruikers  
C. Neggers, Katholieke Universiteit Nijmegen
- Micro/RSX V3.0 en RSX11M-Plus V3.0  
S. Verlee, Digital Equipment b.v. Utrecht
- PRFDAT, een Database systeem bij Hoogovens  
T. Driessen, Pandata Rijswijk
- Networks and Communications: 'The Business Case'  
Ph. de Laubadère, Digital Equipment Europe, Genève
- Planning en voorbereiding om succesvol werkstations in de organisatie te introduceren  
J.W.J. van Till, James Martin Associates, Amstelveen
- Werkstations en informatie-architectuur  
W.L. van Dinten, Rabobank Nederland
- (De)centralisatie  
J.A. de Jong, Koninklijke Marine Den Haag

Neem even contact op met het DECUS secretariaat, tel. 030-83 20 55 / 83 20 89



## De RT-11-SIG op het DECUS Holland symposium

Het thema van het DECUS Holland symposium op 23 en 24 april van dit jaar was, zoals bekend, 'communicatie'. Op de eerste dag waren er 's middags een aantal parallel sessies georganiseerd door de diverse 'Special Interest Groups'.

Het bestuur van de RT-11-SIG had een tweetal sprekers uitgenodigd om het een en ander te vertellen over datacommunicatie en netwerken. Ongeveer 30 DECUS leden maakten van de gelegenheid gebruik om deze lezingen bij te wonen en met de andere aanwezigen van gedachten te wisselen.

De voorzitter opende de bijeenkomst met een aantal vragen betreffende de wens of de noodzaak tot datacommunicatie. De antwoorden toonden aan dat het onderwerp wel degelijk in de belangstelling lag aangezien vrijwel alle aanwezigen binnen het eigen bedrijf of instelling over 4 of meer (DEC) computersystemen beschikt.

### DEC-net/RT

De eerste lezing werd gegeven door Peter Mulder van Digital Equipment en betrof DEC-net. DEC-net lijkt op het eerste gezicht de allesomvattende oplossing voor het data-communicatie-probleem. Tijdens de lezing bleek echter dat er aan DEC-net/RT nogal wat bezwaren kleven en het meestal een vrij dure oplossing is. Jammer genoeg hebben we moeten constateren dat het soort netwerken dat DEC middels DEC-net voor ogen staat nogal verschilt van hetgeen de gemiddelde RT-11 gebruiker nodig heeft.

Een en ander was eigenlijk in de ochtendsessie al duidelijk geworden. Er kwamen daarin een aantal sprekers aan bod die ons zeer imposante netwerken voortoverden met daarin tientallen VAX-en en andere grote systemen (van DEC, IBM of andere origine) die over nagenoeg de gehele wereldbol verspreid kunnen zijn. De mogelijkheden die DEC thans biedt lijken op dat punt schier onbegrensd te zijn.

De slotconclusie die eigenlijk iedereen trok uit Peter Mulder's heldere uiteenzetting was dat het mogelijk is om met DEC-net/RT (als eind-node met beperkte mogelijkheden) aansluiting te vinden bij een groter netwerk, maar dat de onmiddellijke behoeften bij RT-11 gebruikers op een wat ander vlak liggen.

### VTCOM / TRANSF

De lezing van Ap Schouten (DEC vertegenwoordiger in het RT-11-SIG bestuur) maakte duidelijk dat de eenvoudigere soorten van datacommunicatie bij de RT-11 gebruiker meer aanslaan dan DEC-net.

Blijkens de reacties hadden velen daartoe zelf programmatuur ontwikkeld. Sinds versie 5.1 van RT-11 is dat niet meer nodig omdat in de distributiekits thans VTCOM en TRANSF aanwezig zijn. Met die twee programma's is het mogelijk om (vrijwel) zonder extra investeringen de noodzakelijke aansluiting bij de buitenwereld te vinden. De documentatie over deze producten is echter nog minimaal. Door zijn bezoek aan Merrimack, eerder dit jaar was Ap echter in staat om ons wat meer informatie te verschaffen. Zo beschikte hij o.a. over gegevens m.b.t. het door TRANSF gebruikte protocol.

Een saillant detail in de lezing was de mededeling dat VTCOM/TRANSF reeds vijf jaar geleden door een van de ontwikkelaars uit de RT-11 groep (die toch zeker over DEC-net kunnen beschikken) voor eigen gebruik geschreven werd.



## Huishoudelijke deel van de jaarvergadering

Tijdens het (matig bezochte) huishoudelijke deel van de jaarvergadering werd Henk Jas officieel tot vertegenwoordiger van de RT-11-SIG in het DECUS-Holland bestuur verkozen. Hij volgt daarmee Ronald Beetz op die wel in het DH-bestuur blijft maar geen lid meer is van de RT-11-SIG.

Een van de taken van Henk is het vertegenwoordigen van de Nederlandse RT-11-SIG in de commissie die het RT-11 gedeelte van het DECUS Europe symposium (dit jaar in Cannes) organiseert.

Jan Willem Briër, voorzitter RT-11 SIG



## VAX SIG vergadering Decus Holland-dag

Cees de Groot (LHW) geeft een overzicht van KERMIT producten. Deze software, oorspronkelijk afkomstig van Columbia University, is beschikbaar voor zeer vele operating- en computersystemen. Het is ontworpen voor data-communicatie over asynchrone verbindingen. Er kunnen bestanden in ASCII- en binary-formaat mee worden overgebracht. Het is, voor niet commercieel gebruik, vrij beschikbaar.

Aard Langen (THE) behandelt vervolgens het communicatienetwerk op de THE. Na vergelijking van de verschillende technieken en onderzoek naar de te stellen eisen en gewenste functionaliteit is uiteindelijk gekozen voor een broadband netwerk van SYTEK. Hierop zijn momenteel 300 terminals aangesloten. Hij gaat in op de onderverdeling van de totale bandbreedte zoals die voor SYTEK wordt toegepast. Er kunnen conflicten ontstaan bij gebruik van andere broadband netwerken over dezelfde fysieke kabel.

Kees Neggers (KUN) gaat in op de problematiek van EARN (European Academic Network). In Europa heeft IBM een stimulans gegeven om op een grote schaal een netwerk te realiseren waarmee research-instellingen onderling gegevens kunnen uitwisselen. Per land is er een centraal toegangspunt naar het Europese net. Voor Nederland is dat Nijmegen. De communicatie geschiedt op basis van IBM protocollen, maar deze zijn ook geïmplementeerd op diverse andere computer systemen (o.a. VAX/VMS). Er worden Phone, mail en filetransfer functies geboden. Dit gaat op basis van store-and-forward. Koppelingen zullen worden gerealiseerd met netwerken in de VS en Japan. Zo ontstaat er een wereldwijd netwerk. Aansluiting kan men uitsluitend verkrijgen als men zich bereid verklaart om via het eigen computer-systeem computers van anderen aan te sluiten. Wilfred Hartgerink geeft tenslotte een overzicht van de SIG-tape copieerregeling. Naast Peter Kroon (KVI Groningen) gaat nu ook Peter v. Schie (Erasmus Rotterdam) tapes copieëren. Hij neemt de zuidelijke provincies voor zijn rekening.

Henk Stiekema



## Verslag RT-11-SIG bijeenkomst d.d. 27/2/85

De eerste RT-11-SIG bijeenkomst van 1985 was waarschijnlijk voor ieder van de circa 70 aanwezigen een nuttige en aangename dag.

Ap Schouten gaf de aftrap en vertelde ons enkele wetenswaardigheden betreffende RT-11 versie 5.2 dat deze zomer beschikbaar komt. Daarin zullen onder meer de LN03 laserprinter (8 pagina's per minuut) en de PdP-11/84 (Unibus versie van de 11/73 en opvolger van 11/44 en 11/70) ondersteund worden. Nieuwe mogelijkheden zijn ook te verwachten m.b.t. het gebruik van BUP (het kunnen terughalen van een enkel bestand uit een 'backup-set') en verbeteringen in VTCOM/TRANSF en de bijbehorende drivers.

Eveneens nieuw is dat de Ethernet-drivers standaard met RT-11 meegeleverd gaan worden. Om er wat zinnigs mee te kunnen doen is er echter wel een zelf te schrijven gebruikersprogramma benodigd.

En last but not least, er komt een nieuwe snelle Basic interpreter voor RT-11! Het schijnt een afgeleide te zijn van Basic-plus zoals die in RSTS/E aanwezig is.

Als gastspreker was uitgenodigd Dr. Larry Caruthers die reeds vele jaren werkzaam is aan de universiteit van Nijmegen. Larry gaf een bijzonder heldere en informatieve lezing over de programmeertaal C.

De SIG-bestuursleden Bert de Geus, Henk Jas en Harry Haenen hielden vervolgens lezingen over respectievelijk de programmeertaal Dibol, Coprocessoren (KXT-11) en programmeerhulpmiddelen.

De dag werd afgesloten met een vraag en antwoord-sessie. Daarin werd onder meer de suggestie gedaan om in de toekomst eens aandacht te schenken aan Graphics. Het bestuur zal trachten om op de RT-11 dag dit najaar daar gevolg aan te geven.

Tijdens de rondvraag werd door het bestuur aan de deelnemers gevraagd of de RT-11 dagen in de huidige vorm aan hun doel beantwoorden. Na enige discussie was de conclusie dat dat inderdaad het geval is en er geen behoefte bestaat aan een ingrijpend andere opzet. Blijft natuurlijk het feit dat er altijd ruimte is voor verbetering, het bestuur ontvangt met graagte voorstellen daartoe c.q. suggesties voor onderwerpen van lezingen.

Jan Willem Briër, voorzitter RT-11 SIG





# Oproep aan de RT-11-SIG leden

Een van de moeilijkste taken van de SIG-bestuurders is het telkens weer vinden van sprekers voor SIG bijeenkomsten. In de praktijk blijkt dat de bestuurders veelal zelf opdraaien voor het geven van een lezing.

Toch weten we zeker dat er voldoende mensen zijn met een zekere ervaring of bijzondere kennis van een bepaald onderwerp. Echter, wij kennen niet iedereen en weten ook niet precies waarvoor men zoal zijn computer gebruikt. De RT-11-SIG is daarom op dit moment bijzonder geïnteresseerd om in contact te treden met leden die ervaring hebben in een van de volgende onderwerpen:

- \* Computer graphics
- \* Bijzondere real time toepassingen
- \* Programmeertalen / programmeerhulpmiddelen
- \* A/D conversies

Een ieder die denkt hierover iets te kunnen vertellen in een lezing van 15 – 45 minuten (ook van andere SIG's dan RT-11) wordt bij deze uitgenodigd contact op te nemen met ondergetekende of met het DECUS secretariaat.

Jan Willem Briër, voorzitter RT-11 SIG



## Nieuwe programma's in de programmabibliotheek

CPM-179	SIG/M vol. 17, Miscellaneous CP/M utilities
CPM-180	SIG/M vol. 18, Miscellaneous Utilities
CPM-181	SIG/M vol. 19, PASCAL Z UG nr. 1 Miscellaneous
CPM-253	CP/M Games
CPM-254	CP/M Utilities nr. 1
CPM-255	CP/M Utilities nr. 2
CPM-256	CP/M Printer Utilities
CPM-257	CP/M Catalog, Archive and Spelling programs
CPM-258	Newspaper Morgue Database
PRO-127	BRASE: A small Database program
PRO-130	STRESS-11: A structural Analysis Program for RT-11
PRO-131	FSTATS: Statistical Analysis Package
PRO-132	RUNOFF M02.4H for P/OS V2
PRO-133	Astronomical Ephemerides
RB-101	DTC/PC : Desktop Calendar for MSDOS on the Rainbow
RB-102	FIDO V10G and Utilities
DM-107	PASCAL-OS/8
M11-101	CGL to ReGIS VT240 Converter

11-SP-80	Best of 82 : RSX SIG tapes evaluation
11-774	RESETV: reset Version file
11-777	MULPLT : A Multiple File Plotting program
11-778	ROMFIX : Modifies an Intel-format program
11-779	FALOUT : Radiation exposure estimator program
11-780	DATASHEET : A screen entry database program
11-781	TEK4EDIT : Full screen editor for Tektronix 4010, 4012 terminals
11-782	RENUM : Renumbering of FORTRAN Labels
11-783	HP26EDIT : Full screen editor for Hewlett-Packard 2647, 2648 terminals
11-784	MCE/DCE CLI Emulator
11-785	VTLIBR : VT 100 Library
11-786	PARLEZ Communication Package
11-787	CD : DR11-W Links Communications software
11-788	VRTARY : Virtual Array Access Routines for RT-11 and TSX-Plus
11-789	Extend your old RT-11 Basic
11-790	BLASIC : Block structured BASIC pre-processor
11-791	SECUR : A DIBOL Subroutine Developed to provide an extra level of security
V-SP-36	PC-8088 Collection nr. 3
V-SP-37	PC-8088 Collection nr. 4
VAX-114	ReGIS DEC-RITE
VAX-115	ReGIS Data Plotting Package
VAX-116	Productivity Tools Demonstration Package
VAX-117	Business Valuation System
VAX-118	CERBERUS : A package to enable the VMS system to temporarily grant privileges to non-privileged users
VAX-119	PASCAL development software
VAX-121	LA100HCBS : LA100 CALComp Library
VAX-122	TCOPY : A high speed tape copy program
VAX-124	DYDRIV, DLDRIV : An RT-11/VMS file transfer utility
VAX-126	DR-IIW VMS/RSX/RT Cornucopia
VAX-127	AKCOUNT : A VMS System Accounting Package

### Symposium Tapes:

11-SP-76	Symposium tape from the RT-11 SIG Fall 1984, Anaheim
11-SP-77	Symposium tape from the RSX SIG Fall 1984, Anaheim
11-SP-78	Symposium tape from the RSTS/BASIC SIG Fall 1984, Anaheim
11-SP-79	Symposium tape from the European RSX SIG, Fall 1984, Amsterdam
V-SP-38	Symposium Tape from the RSX SIG, Fall 1984, Anaheim in VMS/BACKUP
10-SP-8	Symposium tape from the TOPS-10 SIG, Fall 1984, Anaheim
20-SP-7	Symposium tape from the TOPS-20 SIG, Spring 1984, Cincinnati

Informatie betreffende distributie-media, vereiste configuratie e.d. kan worden verstrekt door het DECUS secretariaat, tel. 030 – 83 20 55 / 83 20 89.





## Nieuwe VAX SIG Tapes. Waar blijven uw bijdragen?

Onlangs hebben wij via Michael Rotert, de Europese VAX SIG tape coordinator, de DECUS VAX SIG tapes van het Fall 1984 Symposium te Anaheim U.S. ontvangen.

Voor de nieuwkomers: deze tapes zijn het resultaat van bijdragen die DECUS-leden kosteloos ter beschikking stellen aan de DECUS-gemeenschap. Deze symposium tapes worden opgenomen in de DECUS library en u kunt ze langs die weg bestellen. Er is echter ook een uit vrijwilligers bestaand internationaal distributie netwerk (zie de Pageswapper), zodat u DECUS Symposium tapes dichterbij huis kunt vinden.

De U.S. Fall 1984 tapes werden begeleid door een briefje van Michael Rotert waarvan hier een uitsnede volgt:

'Please keep in mind that we didn't have a European SIG-tape at Amsterdam. So you should ask your members NOW for submissions to produce a CANNES tape. I will definitely not go on delivering the tapes free if I do not get enough submission to produce a European tape'.

Sinds het bestaan van de Nederlandse VAX SIG zijn door vrijwilligers gratis ca. 150 kopieën van de symposium tapes in Nederland gedistribueerd. Het totaal aan Nederlandse bijdragen aan Europese VAX SIG tapes is minder dan een kwart 600 ft magneetbandje. Het wordt dus hoog tijd dat u eens nagaat wat er tussen uw producten en brouwsels mogelijk interessant zou kunnen zijn voor anderen. Ons enthousiasme om een slechts in één richting werkend distributienet in stand te houden neemt, met dat van Michael Rotert, langzaam af.

De nieuw beschikbare tapes zijn:  
VAXSIG tape nr. 10 en nr. 11 met daarop VAX84C en VAX84D, dat zijn deel 1 en 2 van de U.S. Fall 84 Symposium bijdrage.

Voor het bestellen van deze tapes zie DECUS Holland Bulletin nr. 25. Wij hebben onlangs de distributie binnen Nederland over twee adressen verdeeld:

Distributie voor Groningen, Friesland, Drente, Overijssel, Gelderland, Utrecht en Noord-Holland:

Peter A. Kroon  
Kernfysisch Versneller Instituut  
Zernikelaan 25  
9747 AA Groningen  
tel. 050 - 11 57 37/ bgg 11 57 00

Distributie voor Zuid-Holland, Zeeland, Brabant, Limburg en de Polders:

Peter van Schie  
Erasmus Universiteit  
Computer Instituut Woudenstein  
Postbus 1738  
3000 DR Rotterdam  
tel. 010 -52 55 11 tst. 3342/3333

## Over TRAMP gesproken.....

Tijdens de RSX-SIG van 28-11-1984 heeft de heer Beetz van Organon een lezing gehouden over het programma TRAMP (DECUS programma 11-692). Aan de hand daarvan hebben wij dit programma aangeschaft, maar het blijkt op onze PDP11 configuratie niet te werken. Tijdens die lezing hebben wij, mijn collega de heer H.J.W. Keeris en ik, twee vragen aan de heer Beetz gesteld, waarvan de eerste was of het programma ook op een PDP 11/40 zonder FPU (Floating Point Unit) en zonder EIS (Extended Instruction Set) zou kunnen draaien. Daarop werd bevestigend geantwoord. De tweede vraag was of TRAMP onder FORTRAN IV kon werken, waarop zijn antwoord was dat het nu wel gecompileerd was met een FORTRAN IV/PLUS compiler, maar dat het op een FORTRAN IV compiler ook zou moeten lukken. Dit blijkt niet waar te zijn, want in verschillende FORTRAN files wordt gewerkt met variabel FORMATTING en dit behoort alleen tot de mogelijkheden van FORTRAN IV/PLUS. Een oplossing hiervoor is wellicht te vinden, maar het is voor ons niet mogelijk om hier op korte termijn tijd voor uit te trekken. Tevens is het onduidelijk hoe na compilatie de \*. OBJ files aan elkaar gelinked zouden moeten worden, want voor dezelfde taak zijn meerdere BUILD-files voorhanden. Wij houden ons graag voor commentaar beschikbaar!

L.J.M. Wubben, tel. 070- 75 77 06

H.J.W. Keeris, tel. 070- 75 75 92

## Oproep aan IAS gebruikers

Als IAS gebruiker heb ik de afgelopen jaren uitgebreid ervaring opgedaan met IAS V3.1. Bij het bedrijf DSM Limburg BV, waar ik werkzaam ben, draaien nu zo'n 25 systemen onder dit operating systeem. De problemen die wij zijn tegengekomen bij de invoering van IAS V3.1 en de bijbehorende layered products zijn legio geweest, maar wij kunnen de systemen nu redelijk goed aan het draaien houden. De nieuwe release IAS V3.2 zal echter ongetwijfeld weer voor de nodige nieuwe problemen zorgen.

D.m.v. deze oproep, waarbij ik u uitnodig met mij contact op te nemen, wil ik proberen na te gaan in hoeverre een snellere uitwisseling van ervaringen, problemen en oplossingen tussen IAS gebruikers te realiseren is, in de hoop daarmee een aantal doublures in werkzaamheden te voorkomen.

Op dit moment ben ik reeds in het bezit van een notitie over IAS V3.2 van de heer Centmayer uit München die de IAS activiteiten in Europa coördineert. Deze notitie heeft hij gestuurd naar alle hem bekende gebruikers. Ook u kunt van zijn ervaringen profiteren. Het enige dat hiervoor nodig is, is een telefoontje of brief naar DSM Limburg BV  
t.n.v. Hans Plasman  
afd. Informatie Services  
Postbus 900  
6160 MJ Geleen  
tel. 04494-6 67 55



# AI Tools and Techniques: R1, XSEL and PTRANS

Excerpted from a lecture given at the DECUS Japan Symposium by John McDermott, Department of Computer Science, Carnegie-Mellon University, Pittsburgh, Pennsylvania.

My purpose is to give some indication of the variety of task domains in which AI tools or knowledge-based programming can be valuable, to indicate some of the reasons why, and some of the things that make AI techniques valuable in these three different domains. As examples, R1, XSEL and a program called PTRANS, will be introduced.

First, a simple picture of Digital's sales and manufacturing process. The order entry stage involves sales people meeting with customers, determining what they need and then placing that order. A configuration stage follows, in which the order is checked for clarity. An order administration and scheduling stage is next, when Digital determines when the customer can be promised the order. Finally, the final assembly and test phase takes place. From there the system is shipped to the customer.

Recently, Digital has tried to avoid sending all of its systems through a final assembly and test stage and, instead, is implementing what they call POM (Point of Manufacturing) distribution, so that orders are sent to the customers directly from the volume plants.

## XSEL's Task

Five years ago when Digital first approached Carnegie-Mellon to talk about the possibility of developing an AI tool, the system configuration problem was discussed first. That problem has been solved with the system called R1 or XCON. Digital was quite happy with the results of that system, so the next system Carnegie-Mellon worked on was one that would help with the order entry task, called XSEL, or the salesperson's assistant.

XSEL's task is to determine the customers's needs, then define a generic system that satisfies those needs and, finally, map the generic system into Digital's product offering. Essentially, the XSEL needs to be able to ask questions of the user or help the salesperson elicit information from the customer. On the basis of the information that is collected, XSEL can specify how many megabytes of disk are required, how much memory, how many terminals, what kind of printing capability, and then XSEL decides what Digital components would best satisfy that description.

What is it about XSEL's task that makes the use of AI technology appropriate? The interesting aspects of XSEL's tasks are two-fold. First of all, it is difficult for XSEL to determine what questions to ask the customer, because customers vary a great deal in the degree of sophistication and knowledge of computers and components needed. Secondly, it is important for XSEL to be able to explain very succinctly what reasons led it to come up with the particular description of needs that it will eventually identify.

The reason that artificial intelligence techniques are valuable in these two situations is that, in the first place, there is a tremendously large number of questions that might be asked, but a salesperson cannot bore a customer with 10,000 questions. Therefore it is important to select

an appropriate and adequate subset of questions that the customer will have answers for and which will allow his or her computing needs to be easily determined.

If a system has a lot of knowledge about the kinds of questions that can be asked and the conditions under which those questions are appropriate, then it is possible for the system to recognise what set of questions are appropriate and to avoid questions that will not be informative.

Secondly, because of the way knowledge-based systems operate, the reason for any step that they take can be read off of the situation in which that knowledge is applied, making an explanation very easy. An explanation consists of pointing to the set of cues that evoke certain questions so that a customer can be told why certain questions had to be asked.

R1's task is very different from XSEL's. It determines whether an order is complete and then defines the spatial relationships among the order's components. These two steps are done in parallel; R1 tries to define the spatial relationship, and it discovers at the same time that some component or other is missing. It then simply adds that component to the order and continues to configure the component.

What is it that makes R1's task an appropriate task for AI? As opposed to XSEL's tasks, (where the knowledge-intensive part of the task is figuring out what questions to ask), in R1's case the problem is to recognise which features of the current partial configuration imply what next step. R1 starts with the set of configurable components in its global working memory, and, on the basis of the set of components that are there, recognises that certain subsystems can be built. Building up those subconfigurations triggers more recognition of what kinds of things can be added, until it has put all of the pieces together.

R1 has to determine how to impose an ordering on the subtasks that minimise the likelihood of its having to backtrack. One of things that happens when people do this task is that they go forward for awhile and then recognise that they are pursuing a path leading to an inappropriate configuration. Then they backtrack, taking a few things apart and starting over again. It is important to do things the right way the first time so that no backtracking has to take place.

The third system, PTRANS, is two systems, IMAX and ILOG. PTRANS does two things: it determines when and where on the floor to build each system, and it tracks the progress of each system so that problems can be resolved as they arrive.

What is it that makes PTRANS' task interesting and why is it that AI tools are appropriate for its task? The interesting thing about PTRANS is that the task of scheduling and determining what things should happen when is straightforward. A final assembly and test plant is really a flow shop. There are not hard scheduling problems to be resolved; the systems typically go through five different stages; people operate on those systems, and so, if one lived in a world in which resources were plentiful and always available, the task of scheduling the construction and shipment of these systems would be trivial. The problem is, that in the world of computer manufacturers, awkward things happen.



First, customers have a habit of changing their minds and submitting change orders. Secondly, the resources that are available, the components, and the availability of resources in the plant is not always what it is supposed to be. Because various things can go wrong, initially made plans are not always implementable. IMAX and ILOG spend time scheduling the orders, but most of their knowledge is in recognising when plans are beginning to go wrong, and the rest of knowledge is in changing these plans so that they once again become implementable.

By making very small changes very frequently, PTRANS keeps the management of the plant or the management of the distribution process in control, because the management sees what minor adaptations need to be made in order to keep the plans for system building and system distribution under control.

Take an example of an R1 order with line items. The line items are entered into R1. R1 explodes these bundles of configurable components down to the configuration level, so the nine different line items with different quantities could eventually become a hundred or so configurable components. The first thing R1 does is look at the 100 configurable components.

It then goes through the process of defining the spatial relationships among all of those configurable components and adding components. It then produces some output. For example, the first page would simply reproduce the input with a few slight modifications, but it would list the nine line items, giving an indication of what each is and then, in the case where it cannot configure some of those line items for whatever reason, it indicates that as a comment.

The next thing it does is show what components it had to add in order to make the order complete. The assumption is that the customer has specified the functionality that is needed, but what R1 recognises is whether there is enough cabinet space, box space, backplane space, whether the cabling is appropriate and so on.

It then adds a unibus expander box, cable backplanes, a unibus adaptor because there are more async modules on the order than are permitted on one bus, so the system has to be expanded to a second unibus, a couple of cabinets, and a terminator. And then a disk drive and a tape drive are replaced, since those on the order were not configured appropriately.

After producing the list of the items that were added, R1 prints out pages depending on the size of the order and the size of the system that is being configured, and prints out a description of the various containers and what their contents are.

A few pages describe the cabling so that R1 indicates what cable is needed, how long that cable should be for all the devices that are connected. If information about the floor layout has been entered, then R1 can indicate what an appropriate placement of components on the floor would be, and it has a page on vectors and addresses. There are various pieces of information that are helpful to anyone actually physically assembling the system.

### **Building Expert Systems**

To explain the five stages that one goes through in building an expert system, R1 will be used as an example system.

The first stage is called 'problem selection', where it is determined whether a problem is appropriate and not too

difficult for an AI technique. Is the problem ill-structured enough to make AI techniques worthwhile, but not so ill-structured as to be intractable?

The next stage is 'system architecture selection' in which a decision is made on which specific AI tool will be used. Then a 'prototyping' or 'shell-building' stage follows, then a 'knowledge acquisition' stage, and finally 'evaluation and continued development'.

### **How was R1's problem selected?**

Through the 1970s Digital had a problem with unibus module configuration. Because the unibus is so unconstrained, the placement of modules is not highly constrained on the unibus. During the 1970s many customers were encountering data-late errors because modules had been placed inappropriately, and so DEC made three attempts to solve the unibus configuration problem or the general configuration problem using traditional programming languages, and none of those three attempts was successful.

I think that the reason that they failed is that the unibus or the configuration problem is in fact ill-structured enough so that traditional tools are not really appropriate. Traditional programming languages assume that there is quite a bit of sequentiality and not much conditionality, or branchiness in the solution to a problem. In the case of the unibus configurer, it takes R1 typically about 1,000 steps to configure a system, and at each one of those steps it has, on the average, three different paths that it can go down.

That is a lot of conditionality from a traditional programming point of view, though, in absolute terms, it's not a lot of conditionality. One of the people at Digital suggested that Carnegie-Mellon University might want to get involved in that problem and see whether, by using some AI tools, the problem could be solved.

### **Digital Equipment and Carnegie-Mellon**

We went to Digital in late 1978, to discuss to what extent it would be possible to use AI tools in solving the configuration problem. The focus of the meeting was to decide whether in fact it should be PDP-11s or the VAX-11/780, which had just been announced, that should be the focus of the configurer.

The argument for PDP-11s was that the PDP-11 configuration problem was really serious. There were two things that made the PDP task particularly hard. One was the variety of components supported on PDP-11 was huge; there are 100,000 components in Digital's price book, so a good deal of information would have to be gathered before a configuration system could be helpful.

Secondly, since the volume of the PDP-11 systems was very high, if we were to build a configuration system to solve that problem, it would have had to have been quite good almost immediately. On the other hand, the pros and cons of developing a configurer for the VAX 11/780, were different since the variety and volume components that were supported at the time was very small.

It was not clear to me at the time that one or other of those choices was far superior to the other one. I remember being perplexed about which system to target the configurer for, and we finally decided that we would do it on the 780, but was not clearcut decision. In retrospect, that seems to me to be amazing. I cannot imagine that I could have been so stupid as to not see the 780 was obviously the system for which we had to develop a configurer.



The 780 turned out to be a very lucky choice. I think that the reason that R1 is the first successful AI system has to do with the fact that the size of the problem was just perfect. The initial problem was really quite straightforward, but the problem that R1 can solve now, is really hard. It is a knowledge-intensive task, and so we had an opportunity to start at a simple level and yet bring a lot of knowledge into the system as its capabilities improved.

Now how would one go about estimating the difficulty of task? With respect to R1, there are a number of different indicators of difficulty. First of all, the number and complexity of the objects that have to be dealt with. So the number of components that R1 has to know about is the number of components that are supported in the systems that it configures.

Currently R1 has descriptions in its data base of about 9,000 components. When it started, all it needed was about 400 components in order to be able to configure the VAX-11/780 when that product was introduced. There are typically about 50 to 150 configurable components per system, 50 for the low-end systems, (the PDP-11/23 + and so on) and 150 for the high-end systems (780, 785). For each component, there are somewhere between 25 and 125 pieces of information that are relevant to configuration. That actually turns out to be quite a bit of information. If on the average there are 50 pieces of information and 100 components, then for a typical order, you have 5,000 pieces of information.

With respect to the nature of the search there are about 100 steps in configuring an average system, so, to configure a 780, it is about 2,000 steps. To configure some of the low-end systems, it is about 500 steps. And then there is an average branching factor or a fan-in and fan-out of about 3, at each step, there are three different places you can go, depending on the nature of the circumstances that you are in, and there are about three different places that you can have come from. It is a fairly complex graph of possible paths.

The effort that was expended in doing the problem selection for R1 consisted of a lot of pain at Digital Equipment Corporation during the 1970s from the configuration problem and then a half-day meeting between two domain experts and two AI researchers. One of the domain experts presented a detailed overview of the PDP-11 configuration task. As a result, Digital offered to provide CMU with access to an expert and CMU decided to try to develop a prototype configurator for 780 systems.

The second stage is the selection of an architecture. It is not clear to me that anybody knows much about how to select an architecture. People who build AI systems tend to become familiar with a particular tool fairly early on, and make that tool the one that they use in building a wide variety of systems. In general, all of the tools that are available, the knowledge representation languages, general purpose production systems, production system architectures are all adequate for just about any kind of ill-structured task. I do not think that there is anyone in the AI community who can discriminate among the suitability of various tools for various kinds of problems.

We used OPS4 for the configuration task because we had been experimenting with OPS for two or three years prior to that, and we wanted to see whether it was a good tool to use to build an AI application system. We were using it in some sense to explore its strengths and weaknesses.

One of the things that attracted us to the configuration task was that it appeared to be one in which we could exploit the strongly recognition driven nature of OPS, so there was a kind of special suitability from our point of view in the configuration task for a general purpose forward chaining production system language.

A rule-based programming language has a rule memory, a working memory and an interpreter. The interpreter works with rules: in R1's case, the number now is about 3300. There are 3300 rules looking-down into a set of descriptions in working memory, and in R1 there are about typically on the average 500 of these descriptions in working memory at any given time.

If the current subtask is assigning devices to unibus modules, and there is an unassigned dual port disk drive and the type of the controller it requires is known, and there are two such controllers neither of which has any devices assigned to it, and the number of devices which these controllers can support is known, if all of those conditions are satisfied, then assign the disk drive to each controller and note that each controller supports one device. That is the kind of rule that typically found in R1.

Several years of effort were expended in developing the OPS language at Carnegie Mellon University. The AI researchers who helped select the configuration task made their evaluation of the suitability of the task with OPS in mind. The result was that OPS was selected as the system architecture before the configuration task was selected as the task to tackle.

### Selecting an AI Tool

It is certainly the case that there are many other AI tools that could be used to develop a configurator. One of the things that gives OPS5 its current edge is, due to the BLISS implementation of OPS5, it is faster than some of the tools that are implemented only in LISP. So, if one were to use the LISP implementation of OPS5 for example, it would take a very long time to configure a system and it would be hard to use the LISP OPS5 interpreter as a production version of the system. The fact that the BLISS OPS5 system is about ten times faster than the LISP system makes it possible to use it effectively.

It is the raw speed of OPS5 that give it an edge currently. As faster hardware is introduced, the edge that OPS5 has in speed is going to decrease and, over time, tools that provide a richer environment, particularly a LISP-based environment, are probably going to supersede OPS.

Stage three is developing a prototype. Our discussions with the domain experts had made it clear to us that there was quite a bit of structure in the configuration task. Although AI tools are more suitable for ill-structured tasks, there are tractable ill-structured tasks and untractable ill-structured tasks. What we need to find at this stage in AI are ill-structured tasks that still have enough structure so that we can begin to understand how to deal with ill-structuredness.

One of the things about the configuration task is that it appeared to consist of 100 or so fairly well defined subtasks that had simple relationships to one another. All of the conditionality appeared to be buried inside each one of those subtasks.

Within each subtask there was ill-structuredness, but the subtask organisation itself was fairly structured. We wanted to exploit that structure that structure so that we would not make the problem harder than it was.

Secondly, the obvious way to approach the task was to just put these components into working memory and then



build up a configuration incrementally. It was a question of taking pairs of configurable components, deciding what their relationship to one another should be, and then continuing to add components until finally all of the components were put into the structure.

Finally, everybody involved in AI knows that backtracking is an inevitable activity. We expected R1 would recognise that it was misconfiguring a system and then backtrack to a point where the configuration was acceptable and go down another path. We assumed that a good deal of R1's knowledge would be in enabling it to recognise its misconfiguring and therefore recognise when to back up.

What were the initial sources of knowledge? R1 had a single domain expert that we dealt with and in addition to that domain expert, we had two VAX 11/780 configuration manuals. When our domain expert left after three months, we got another expert. The two manuals that we had were written by still other people. And so, from the very beginning, we had a variety of sources of expertise. The problem with having only one source of expertise is that the expert can be strongly focused on a particular set of sub-issues, and fail to give an adequate overview of the task.

The early version of R1 recognised which features of the current partial configuration imply what next step and it determined dynamically how to impose an ordering on the subtasks that minimised the likelihood of having to backtrack. R1, from the very beginning, brought a lot of knowledge to bear which reduced tremendously the amount of search.

After four months, R1 could configure the simplest of Digital's standard VAX 11/780 package system, but it immediately ran into trouble if given a more complex system to configure. This version of R1 knew all the basic kinds of activities required to configure a system, so it was initiated, but basically it knew almost nothing.

**The amazing thing about that version of R1** and all of the subsequent, more knowledgeable versions of R1, is that it really did not have to do any significant backtracking. The system is characterised by not doing any backtracking. It avoided backtracking by not doing any search, but it did its search before making a move so that it collected all the information it needed in order to be quite confident that its next step was appropriate. Then it would make that step, and if it saw that it did not have quite enough knowledge to make the next step, it would go out and collect or generate more information. It engages in behaviour which I would appropriately call 'look around'.

R1's personality is kind of tedious; it is not a risk-taking system. Rather, it carefully produces enough information so that it can always have a high degree of confidence in its judgment. But the fact that it is possible for R1 to produce that kind of information, and collect enough knowledge which will push the system forward down the right path so that backtracking is unnecessary, is an exciting thing.

In fact, that is where we should expect knowledge-based programming to lead us. What we say is that knowledge is a tremendously powerful tool for avoiding search or avoiding backtracking. What we see in R1 is that taken to an extreme. The fourth stage is knowledge acquisition. R1 acquired its knowledge just the same way that any self-respecting expert system acquires its knowledge. The prototype of R1 was given a set of components to configure. An expert was shown R1's output, and if the expert identified a problem, he was asked to indicate what knowledge R1 must be missing in order to make that

mistake. The expert educated the knowledge engineer about the knowledge that R1 must be missing in order to make that kind of mistake, and then the knowledge engineer formulated one or more rules and added them to R1. Then the prototype was given another set of components to configure and the process just repeated itself.

After three months, R1 had three times as much configuration knowledge as it had had after four months. After four months, it had had 250 rules; after ten months, it had 750 rules. It could correctly configure most of the systems it was asked to configure irrespective of their complexity. And, in fact, it impressed the experts.

What can we say about R1's knowledge acquisition stage? The effort expended during R1's knowledge acquisition stage was five months by a knowledge engineer to build up R1's knowledge base, one month of effort by a domain expert reviewing R1's output and supplying missing knowledge, and one month of effort by a domain expert building up a data base of component descriptions.

The result: a version of R1 that surprised the experts with its ability to deal correctly with highly unusual situations.

In May of 1980, OPS5 became available. We decided that we would re-represent the knowledge in R1 that had been represented as OPS4 rules as OPS5 rules.

The difference between OPS4 and OPS5 was not terribly great. We could have automatically translated the rules from one form to the other, but we looked upon it as an opportunity to see whether there are a lot of generalisations that were missed by the incremental addition of knowledge. As you would expect, there were so that, at the end of this re-representing phase, R1 had about 500 rules, so it had two-third as many rules as it had had and it was more competent. We had simply found more general ways of representing the knowledge. The focus of the work on R1 then changed, from CMU to Digital Equipment Corporation, so from the point that R1 had 500 rules up until the point that it had 3300, (January 1983) most of the work on R1 took place at Digital Equipment Corporation. During that time, R1 was extended so that as each VAX was introduced, it was able to deal with that VAX.

In 1982, it was extended to deal with its first PDP-11, and now R1 can configure all of Digital's products that are reasonably high volume.

Although it might appear that most of the knowledge that was added to R1 was added so that it could deal with these new systems, in fact that is not the case. Only about 35 percent of knowledge that was added, was added in order to deal with new systems. The rest of the knowledge was knowledge needed in order to deal with situations that had not been previously encountered.

Is this rate of addition of knowledge going to keep on going forever? Considering the number of orders that R1 has configured over the last several years after the first year, about 600 orders, after the second year 8,000 after the third year about 3,500, after the fourth year, over 80,000 orders, and by now well over 150,000, the rate at which it is encountering new situations is growing very fast. But the growth in knowledge is linear so that it needs to encounter many more situations now before it finds one that it is unfamiliar with. As the number of systems that R1 configures each quarter begins to level off a little bit, its growth in knowledge is going to level off even more.

---

Dit artikel werd overgenomen uit Leaders Digest, Vol.3, No.3, mei 1985



# Colofon

## Redactie

W.P. Ingenegeren  
Rijksuniversiteit Utrecht  
Exper. Fysika  
Postbus 80000  
3508 TA UTRECHT  
Tel.: (030) 53 14 98

J.P. Hamaker  
Radio Sterrenwacht  
Oude Hoogeveensedijk 4  
7991 PD DWINGELOO  
Tel.: (05219) 72 44

J. Smeenk  
Ideta  
Postbus 12644  
1100 AP AMSTERDAM  
Tel.: (020) 5 90 42 16

W.G. de Geus  
Minihouse B.V.  
Postbus 1066, 2803 PL GOUDA  
Tel.: (01820) 6 29 11

## DECUS Holland Bestuur

Dr. R. Beetz, voorzitter  
Ir. K. Lingbeek EDC vertegenwoordiger  
H. Jas  
T. Driessen, sekretaris  
W. Hartgerink, penningmeester

## Korrespondentie-adres DECUS Holland

Digital Equipment Computer Users Society  
Postbus 9212  
3506 GE UTRECHT  
Tel.: (030) 83 20 55

## Korrespondentie-adres DECUS Europe

Digital Equipment Computer Users Society  
P.O. Box 510  
CH-1213 PETIT LANCY/GE  
Zwitserland  
Tel.: (41) (22) - 93 33 11

## Program Library Coördinator

G. Goris  
Arsycom  
Kabelweg 43  
1014 BA AMSTERDAM  
Tel.: (020) 82 38 58

---

**Sluitingsdatum kopij voor  
DECUS HOLLAND BULLETIN  
NUMMER 27:  
25 SEPTEMBER 1985**

---

## SIG Adressen

**RTII SIG**  
J.W. Briër  
Datelcare  
Huis ter Heideweg 28  
Postbus 2  
3700 AA ZEIST  
Tel.: (03404) 2 13 44

**RSX SIG**  
T. Driessen  
Pandata BV  
Verrijn Stuartlaan 28  
2289 EL RIJSWIJK (ZH)  
Tel.: (070) 94 93 25

**RSTS SIG**  
Dr. L.K.J. van Romunde  
Erasmus Universiteit  
Afd. Epidemiologie  
Dr. Molenwaterplein 50  
3015 GE ROTTERDAM  
Tel.: (010) 63 44 65

**PC SIG**  
Ir. K. Lingbeek  
Landbouwhogeschool Wageningen  
Hollandseweg 1  
6706 KN WAGENINGEN  
Tel.: (08370) 8 37 78

**VAX SIG**  
E.W. Hartgerink  
I.T.C.  
350 Boulevard 1945  
7511 AK ENSCHEDE  
Tel.: (053) 32 03 30

**MANAGEMENT SIG**  
J.A. de Jong  
Koninklijke Marine  
Gebouw Admiraliteit  
v.d. Burchlaan 31  
2597 PC DEN HAAG  
Tel.: (070) 16 24 85

The following are trademarks of DIGITAL Equipment Corporation.

DEC	DECnet	IAS
DECUS	DECsystem-10	MASSBUS
Digital Logo	DECsystem-20	PDT
PDP	DECwriter	RSTS
DIBOL	Word Processor	VMS
RSX	Ergodynamic	VT
VAX	Professional	LA100
LA50	Softsense	EduSystem
DECmate	RAINBOW	P/OS
LQP02		

CP/M is a Trademark of Digital Research, Inc.

